Terms used
register and **'execution core'** and **'system call instruction'** and **flag** and **mask** and **'operation mode'**

Found **2,857** of **134,837**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 Early load address resolution via register tracking

Michael Bekerman, Adi Yoaz, Freddy Gabbay, Stephan Jourdan, Maxim Kalaev, Ronny Ronen

May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2

Full text available: pdf(143.17 KB) Additional Information: full citation, abstract, references, citings, index terms

Higher microprocessor frequencies accentuate the performance cost of memory accesses. This is especially noticeable in the Intel's IA32 architecture where lack of registers results in increased number of memory accesses. This paper presents novel, non-speculative technique that partially hides the increasing load-to-use latency, by allowing the early issue of load instructions. Early load address resolution relies on register tracking to safely compute the addresses of memory refere ...

2 Instruction path coprocessors

Yuan Chou, John Paul Shen

May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2

Full text available: pdf(134.64 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents the concept of an Instruction Path Coprocessor (I-COP), which is a programmable on-chip coprocessor, with its own mini-instruction set, that operates on the core processor's instructions to transform them into an internal format that can be more efficiently executed. It is located off the critical path of the core processor to ensure that it does not negatively impact the core processor's cycle time or pipeline depth. An I-COP is highly versatile and can be used ...

3 Architecture of the IBM system/370

Richard P. Case, Andris Padegs

January 1978 **Communications of the ACM**, Volume 21 Issue 1

Full text available: pdf(2.78 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper discusses the design considerations for the architectural extensions that distinguish System/370 from System/360. It comments on some experiences with the original objectives for System/360 and on the efforts to achieve them, and it describes the reasons and objectives for extending the architecture. It covers virtual storage, program control, data-manipulation instructions, timing facilities, multiprocessing, debugging and monitoring, error handling, and input/output operations. ...

Keywords: architecture, computer systems, error handling, instruction sets, virtual storage

4 The KScalar simulator

J. C. Moure, Dolores I. Rexachs, Emilio Luque

March 2002 **Journal on Educational Resources in Computing (JERIC)**, Volume 2 Issue 1

Full text available: pdf(493.35 KB) Additional Information: full citation, abstract, references, index terms

Modern processors increase their performance with complex microarchitectural mechanisms, which makes them more and more difficult to understand and evaluate. KScalar is a graphical simulation tool that facilitates the study of such processors. It allows students to analyze the performance behavior of a wide range of processor microarchitectures: from a very simple in-order, scalar pipeline, to a detailed out-of-order, superscalar pipeline with non-blocking caches, speculative execution, and comp ...

Keywords: Education, pipelined processor simulator

5 The evolution of the DECsystem 10

C. G. Bell, A. Kotok, T. N. Hastings, R. Hill

January 1978 **Communications of the ACM**, Volume 21 Issue 1

Full text available: pdf(1.92 MB) Additional Information: full citation, abstract, references, citings, index terms

The DECsystem 10, also known as the PDP-10, evolved from the PDP-6 (circa 1963) over five generations of implementations to presently include systems covering a price range of five to one. The origin and evolution of the hardware, operating system, and languages are described in terms of technological change, user requirements, and user developments. The PDP-10's contributions to computing technology include: accelerating the transition from batch oriented to time sharing computing systems; ...

Keywords: architecture, computer structures, operating system, timesharing

6 Integrated techniques for functional and gate-level digital logic simulation

S. A. Szygenda, A. A. Lekkos

June 1973 **Proceedings of the tenth design automation workshop on Design automation**

Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, citings, index terms

Digital simulation is defined as the process of exercising a realistic software model of a digital system, with sets of input stimuli, to generate signal behavior versus time. Simulation processes can be categorized by their simulation atom. Gate-level simulation 1,2,3,4 is simply one in which the atoms of simulation are gates. Functional simulation 5,6,7,8 simulates larger units, such as adders, decoders, registers, etc. It appears that ...

7 α-coral: a multigrain, multithreaded processor architecture

Mark N. Yankelevsky, Constantine D. Polychronopoulos

June 2001 **Proceedings of the 15th international conference on Supercomputing**

Full text available: pdf(196.56 KB) Additional Information: full citation, abstract, references, index terms

Recently popularized hardware multithreading (HMT) architectures, such as SMT, Multiscalar and Terra do not provide flexible and efficient methods of thread management and synchronization in hardware. The &agr;-Coral architecture is a tool for investigation of a more dynamic approach to thread management. Unlike other architectures, there are no strict requirements on timing and size of threads, and no static partitioning of resources. &agr;-Coral provides for simultaneous multiprogramming an ...

Keywords: multithreaded, parallelizing compiler, processor archietecture

8 A block-and-actions generator as an alternative to a simulator for collecting architecture measurements

M. Huguet, T. Lang, Y. Tamir

July 1987 **ACM SIGPLAN Notices , Papers of the Symposium on Interpreters and interpretive techniques**, Volume 22 Issue 7

Full text available: pdf(1.12 MB) Additional Information: full citation, abstract, citings, index terms

To design a new processor or to modify an existing one, designers need to gather data to estimate the influence of specific architecture features on the performance of the *proposed machine* (PM). To obtain this data, it is necessary to measure on an *existing machine* (EM) the dynamic behavior of *typical programs*. Traditionally, *simulators* have been used to obtain measurements for PMs. Since several hundred EM instructions are required to decode, interpret, and measure e ...

9 Some requirements for architectural support of software debugging

Mark Scott Johnson

March 1982 **Proceedings of the first international symposium on Architectural support for programming languages and operating systems**, Volume 10 , 17 Issue 2 , 4

Full text available: pdf(710.87 KB) Additional Information: full citation, abstract, references, citings, index terms

Architectural support of high-level, symbolic debugging is described at three levels of abstraction: the user's view of desired debugging functionality, the debugger implementor's view of architectural requirements that support the functionality, and the computer architect's view of architectural features or attributes that implement the requirements. References are made where possible to computing systems that meet the requirements. The paper is written from the viewpoint of debugger imple ...

Keywords: Architectural debugging support, Breakpoints, Debugging, Debugging-oriented architecture, Interactive debugging, Profiles, Symbolic debugging, Traces

10 System-level power optimization: techniques and tools

Luca Benini, Giovanni de Micheli

April 2000 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 5 Issue 2

Full text available: pdf(385.22 KB) Additional Information: full citation, abstract, references, citings, index terms

This tutorial surveys design methods for energy-efficient system-level design. We consider electronic sytems consisting of a hardware platform and software layers. We consider the three major constituents of hardware that consume energy, namely computation, communication, and storage units, and we review methods of reducing their energy consumption. We also study models for analyzing the energy cost of software, and methods for energy-efficient software design and compilation. This survery ...

11 Hardware and software tools for the development of a micro-programmed microprocessor

James Nash, Mike Spak

November 1979 **Proceedings of the 12th annual workshop on Microprogramming**

Full text available: pdf(674.94 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper discusses the development and implementation of a number of hardware and software tools used in the design, development, and debugging of the microcode and nanocode for the MC68000 microprocessor. A functional description of these tools is included as well as an analysis of how each worked with respect to solving the desired problem (ie., generating correct microcode and nanocode). Peculiarities in the integrated

circuit industry for microcode verification is considered as well a ...

12 A building block approach to color graphics

J. Robert Flexer, Gio Wiederhold

August 1979 **ACM SIGGRAPH Computer Graphics , Proceedings of the 6th annual conference on Computer graphics and interactive techniques**, Volume 13 Issue 2

Full text available: pdf(1.21 MB) Additional Information: full citation, abstract, references, index terms

Graphics and imaging are important in scientific, academic and industrial environments. In the past graphics systems have been used with large computers and were only available to a minority of users. The relatively small and specialized use of graphics has inhibited sharing of software and prevented standardization necessary for widespread use. Dense semiconductor memory has recently become easily available in large quantities and makes high resolution graphics and imaging systems feasible ...

Keywords: Color graphics, Frame buffer, Imaging, Lightpen, Photo trigger, Rasterscan display, S-100 bus, Video digitizer, Video display

13 Percola: a special purpose programmable 64-bit floating-point processor

J. M. Normand

June 1988 **Proceedings of the 2nd international conference on Supercomputing**

Full text available: pdf(1.64 MB) Additional Information: full citation, references, index terms

14 SASI enforcement of security policies: a retrospective

Úlfar Erlingsson, Fred B. Schneider

September 1999 **Proceedings of the 1999 workshop on New security paradigms**

Full text available: pdf(862.14 KB) Additional Information: full citation, references, citings, index terms

15 Runtime Power Monitoring in High-End Processors: Methodology and Empirical Data

Canturk Isci, Margaret Martonosi

December 2003 **Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture**

Full text available: pdf(921.50 KB) Publisher Site Additional Information: full citation, abstract

With power dissipation becoming an increasingly vexingproblem across many classes of computer systems, measuringpower dissipation of real, running systems has becomecrucial for hardware and software system research and design.Live power measurements are imperative for studiesrequiring execution times too long for simulation, such asthermal analysis. Furthermore, as processors become morecomplex and include a host of aggressive dynamic powermanagement techniques, per-component estimates of powerd ...

16 A high-speed message-driven communication architecture

J. Peterson, E. Chow, H. Madan

June 1988 **Proceedings of the 2nd international conference on Supercomputing**

Full text available: pdf(1.27 MB) Additional Information: full citation, abstract, references, citings, index terms

The performance of a message-passing multiple instruction multiple data (MIMD) concurrent computer depends in large part on the communication processing overhead. A high-speed communication architecture is proposed for a hypercube-type supercomputer to attain the specific goals of message-driven processing. These goals include: direct hardware execution of messages, queueing of messages (using various paradigms), adaptive message routing, and special local registers for fast context ...

17 Our machine, a microcoded LSI processor

Dave Johannsen

November 1978 **Proceedings of the 11th annual workshop on Microprogramming**

Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, citings, index terms

Current LSI technology allows the systems designer to construct complex data processing structures containing tens of thousands of transistors on single silicon chips. Constraints imposed by the technology influence design tradeoffs and result in computer architectures dramatically different from the more classical computer designs. At CalTech we are exploring the possibilities offered by nMOS technology, with the "Our Machine" (OM) project being one of the current research proj ...

18 Process oriented logic simulation

Sany M. Leinwand

June 1981 **Proceedings of the eighteenth design automation conference on Design automation**

Full text available: pdf(662.44 KB) Additional Information: full citation, abstract, references, index terms

Theoretically, simulation can be activity oriented, event oriented or process oriented. Existing techniques for logic simulation are either activity or event oriented. In this paper, the possibility of logic simulation using process oriented concepts is investigated. Such an approach is justified by the need to support modular design environments. The key feature is that of asynchronous module activity: the timing order of signal changes has to be preserved only for those events belonging t ...

19 Session 4C: Computer architecture: A new simulator workbench for comparing SIMD processing element architectures

Todd C. Marek

April 1992 **Proceedings of the 30th annual Southeast regional conference**

Full text available: pdf(608.48 KB) Additional Information: full citation, abstract, references

The impact of machine structure on system performance is a critical consideration in designing highly integrated SIMD architectures. This issue is affected by PE granularity, PE complexity, and interconnection structure. Detailed analysis of the issues related to this structure/performance relationship is relevant in developing new massively parallel architectures. To meet this need, a simulator workbench which can be used to quantitatively evaluate the performance of a wide range of machine str ...

20 The Postroom Computer

Hugh Osborne

December 2001 **Journal on Educational Resources in Computing (JERIC)**, Volume 1 Issue 4

Full text available: pdf(242.80 KB) Additional Information: full citation, abstract, references, index terms

The Postroom Computer is a computer architecture simulator based on the *Little Man Computer* developed in 1965 by Stuart Madnick and John Donovan. It provides a family of architectures suitable for use in teaching introductory computer architectures. It is designed to introduce aspects of computer architecture and low-level programming in an incremental way. The extensions are designed to provide a range of computing models within the Little Man Computer paradigm. As they are introduced th ...

Keywords: Computer architecture simulator, education

h c g e cf c